(2)

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

A PROTOTYPIC MODEL FOR SCHEDULING COURSES
AT THE NAVAL POSTGRADUATE SCHOOL

by

Wu Hsi-Hsien

December 1993

Thesis Advisor:	R. Kevin Wood
Second Reader:	Richard E. Rosenthal

Approved for public release; distribution is unlimited.

Unclassified
Security Classification of this page

REPORT DOCUMENTATION PAGE

1a Report Security Classification: Unclassified	1b Restrictive Markings
2a Security Classification Authority	3 Distribution/Availability of Report Approved for public release; distribution is unlimited.
2b Declassification/Downgrading Schedule	
4 Performing Organization Report Number(s)	5 Monitoring Organization Report Number(s)
6a Name of Performing Organization Naval Postgraduate School — 6b Office Symbol (*if applicable*) *OR	7a Name of Monitoring Organization Naval Postgraduate School
6c Address (*city, state, and ZIP code*) Monterey CA 93943-5000	7b Address (*city, state, and ZIP code*) Monterey CA 93943-5000
8a Name of Funding/Sponsoring Organization — 6b Office Symbol (*if applicable*)	9 Procurement Instrument Identification Number
Address (*city, state, and ZIP code*)	10 Source of Funding Numbers Program Element No — Project No — Task No — Work Unit Accession No

11 Title (*include security classification*) A PROTOTYPIC MODEL FOR SCHEDULING COURSES AT THE NPSSCHOOL

12 Personal Author(s) WU, Hsi-Hsien

13a Type of Report	13b Time Covered	14 Date of Report (*year, month, day*)	15 Page Count
Master's Thesis	From To	Dec 93	63

16 Supplementary Notation The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17 Cosati Codes			18 Subject Terms (*continue on reverse if necessary and identify by block number*)
Field	Group	Subgroup	Timetable, Time-Table, College Schedules, University Schedules, Operations Research, Scheduling, Integer Linear Programming, NPS Scheduling System

19 Abstract (*continue on reverse if necessary and identify by block number*)

This thesis develops a prototypic integer programming model to aid in solving the Naval Postgraduate School academic course scheduling problem. The simplified model schedules faculty members to teach their assigned courses in specific rooms at specific times and schedules groups of students to the courses they have requested. The model assures, as best possible, that room capacity is not exceeded, students and faculty have time for lunch and faculty requesting "back-to-back" courses are accommodated.

To make the problem managable, we concentrate on just one building, Glasgow Hall, and three departments, Operations Research, Mathematics and National Security Affairs. Even doing this, the model generated in GAMS (Generalized Algebraic Modeling System) has about 287,778 variables and 148,161 constraints and is too large to solve. Consequently, a simplified model, restricted to the Operations Research Department, is solved. This problem encompasses 19 faculty members, 26 courses, 83 sections and 11 classrooms. The model has less than 32,000 variables and 17,000 constraints and is solved using GAMS and the X-System on an Amdahl 8995-700A in 3488.4 seconds

20 Distribution/Availability of Abstract xx unclassified/unlimited _ same as report _ DTIC users	21 Abstract Security Classification Unclassified	
22a Name of Responsible Individual R. Kevin Wood	22b Telephone (*include Area Code*) (408) 656-5676	22c Office Symbol OR/WD

DD FORM 1473,84 MAR 83 APR edition may be used until exhausted security classification of this page
All other editions are obsolete Unclassified

Approved for public release; distribution is unlimited.

A PROTOTYPIC MODEL FOR SCHEDULING COURSES AT THE NAVAL POSTGRADUATE SCHOOL

by

Wu Hsi-Hsien
Major, Chinese Army
B.S., Chung Cheng Institute of Technology, 1984

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
December 1993

Author: Wu Hsi-Hsien

Wu Hsi-Hsien

Approved by: R. K. Wood

R. Kevin Wood, Thesis Advisor

Richard E. Rosenthal

Richard E. Rosenthal, Second Reader

Peter Purdue, Chairman
Department of Operations Research

ABSTRACT

This thesis develops a prototypic integer programming model to aid in solving the Naval Postgraduate School academic course scheduling problem. The simplified model schedules faculty members to teach their assigned courses in specific rooms at specific times and schedules groups of students to the courses they have requested. The model assures, as best possible, that room capacity is not exceeded, students and faculty have time for lunch, and faculty requesting "back-to-back" courses are accommodated.

To make the problem managable, we concentrate on just one building, Glasgow Hall, and three departments, Operations Research, Mathematics and National Security Affairs. Even doing this, the model generated in GAMS (Generalized Algebraic Modeling System) has about 287,778 variables and 148,161 constraints and is too large to solve. Consequently, a simplified model, restricted to the Operations Research Department, is solved. This problem encompasses 19 faculty members, 26 courses, 83 sections and 11 classrooms. The model has less than 32,000 variables and 17,000 constraints and is solved using GAMS and the X-System on an Amdahl 8995-700A in 3488.4 seconds.

TABLE OF CONTENTS

ACKNOWLEDGEMENT

This research could not have been completed without the instruction and encouragement of Professor Wood who corrected my spelling and writing, again and again, with patience and kindness, and helped me with the difficult creation and implementation of the model. I'm very honored to say: Professor Wood is my thesis advisor.

I would like to extend my appreciation to Professor Rosenthal, who contributed his expertise in the field of Optimization and assisted me when I really needed it. Thanks also to NPS Registrar Mr. Tracy Hammond for explaining the current scheduling process, to Miss Helen Davis for helping to solve my SAS problem, to Mr. Richard E. Donat for solving a system memory problem and to Mr. Larry Frazier for instruction on "WordPerfect".

Finally, I'd like to give a special thank-you hug to my wife, Ming-Man. Without her understanding and support, this thesis would not have been possible.

I. INTRODUCTION

This thesis develops a prototypic integer programming model to help schedule academic courses at the Naval Postgraduate School, a task that currently requires many weeks of manual work for each quarter.

A. BACKGROUND

The Naval Postgraduate School (NPS) is atypical compared to most public and private academic institutions with respect to its course scheduling process. A brief description is presented:

Operated under the auspices of the Chief of Naval Operations, the Naval Postgraduate School located in Monterey, California, is the Navy's graduate school. The school's emphasis is on study and research programs relevant to the Navy's interests, as well as to the interests of other arms of the Department of Defense. The programs are designed to accommodate the unique requirements of the military. The school's primary purpose is to offer graduate education to officers of U.S. and allied forces to increase their combat effectiveness.

Nearly 2000 students attend NPS. The student body consists of officers from the five U.S. uniformed services, officers

from approximately 35 allied countries and a small number of civilians. Most study programs lead to Masters degrees and fulfill the requirements for a military occupational specialty code for service members [Ref.2].

Courses are offered each quarter to meet estimated demand. Estimates are fairly accurate because students have few electives and their course sequences are largely fixed. Students sign up for three or four courses and the scheduler tries to arrange the courses and instructors so that every student can take every course he or she requests, with very few exceptions. This is unlike a civilian university where, typically, courses are scheduled at specific times, students sign up for those courses and times, and if the number of students signed up for a course reachs a specified limit, the course is closed to further enrollment.

B. CURRENT NPS COURSE SCHEDULING

The academic year at NPS is divided into four quarters, Fall, Winter, Spring and Summer. Each academic quarter consists of eleven weeks of course work, followed by a one-week period for final examinations. Schedules for classes and examinations are developed independently and primarily through the manual efforts of two class schedulers. Preparing the schedules for an academic quarter requires full-time work from the schedulers during most of the previous quarter. More detail on the scheduling process can be found in [Ref.1].

The scope of the scheduling problem at NPS is enormous. Each quarter over 160 professors and 40 military instructors must be scheduled to teach more than 300 courses to about 2000 students in approximately 100 classrooms and laboratories. NPS tries, more than most schools, to tailor course offerings to student requests. The schedulers must fit the students' requests into a master schedule. They must also accommodate recurring events such as departmental meetings, and constraints resulting from the instructors' requirements and preferences for certain days, time periods and classrooms.

C. THESIS INTENT

The intent of this thesis research is to automate some of the quarterly academic scheduling task at NPS with the use of a prototypic integer programming model. Several attempts have been made over the past 35 years to achieve this aim by automating either part or all of the scheduling process. To date, all attempts to totally automate the NPS scheduling process have failed. [Ref.1]

To make the problem managable, we concentrate on just one building, Glasgow Hall, and the three departments housed in it, Operations Research (OR), Mathematics (Math) and National Security Affairs (NSA). This scheduling problem is largely independent from the rest of the school since most of the OR, Math and NSA courses are taught only in Glasgow Hall. For the

1993 Winter quarter, about 89% of the courses taught by those departments were scheduled in this building.

The Glasgow Hall problem covers 66 faculty members, 562 sections, 81 courses, 17 classrooms, 9 periods and 5 weekdays for Winter 1993 data. In order to obtain some useful results, we simplify the model further to consider only OR, which covers 19 faculty members, 83 sections, 26 courses, 10 classrooms and 7 periods in one weekday, for the same quarter.

The Glasgow Hall and OR Department scheduling problems are formulated as integer programming models in this thesis. These models are then implemented in The Generalized Algebraic Modeling System (GAMS) [Ref.3] and solved using the X-System [Ref.4].

D. THESIS OUTLINE

Chapter II describes the problem, the assumptions that are made and the requirements a model to solve the problem must meet. Chapter III presents the integer programming model. Chapter IV gives computational results. Chapter V gives conclusions and recommendations for follow-on work required to fully implement the design generated in this thesis.

II. PROBLEM DESCRIPTION

A. PROBLEM DEFINITIONS

1. The Academic Calendar

The NPS academic year starts on the Monday nearest October 1st and is divided into four quarters of 12 weeks each (Fall, Winter, Spring, Summer), with 11 weeks of instruction and one week of final examinations. There are two two-week breaks between Fall and Winter quarters in December, and between Spring and Summer quarters in June. Each academic week consists of five days (Monday-Friday) and each academic day has nine one hour periods of instruction. The academic day begins with Period #1 at 0800 and ends with the conclusion of Period #9 at 1700. The first 10 minutes of each period are intended to allow students time to travel between classes and and the last 50 minutes are used for teaching the course. In this thesis, we are concerned only with scheduling courses; scheduling final examinations is a completely different problem.

2. Schedule Elements

The principal elements of any school schedule are students, courses, instructors, classrooms and time periods for instruction. These elements are related to each other in

defined ways, i.e., students take courses, which are taught by instructors, in classrooms, during scheduled time periods. Constructing the NPS course schedule is essentially a matter of solving a number of interconnected, and sometimes conflicting, pairing problems involving these elements (e.g., student-course, course-instructor).

At NPS each quarter roughly 1500-2000 students, grouped in approximately 950 "sections" (students who take the same courses are called a "section"), in 38 curricula within 11 curricular programs, enroll in about 300 courses which are taught by about 200 faculty members from a total of almost 900 courses listed in the course catalog [Ref.2]. Some courses are divided into one or more "segments" to accommodate more students in a course than seating space in classrooms allow, and to maintain an acceptable student-instructor ratio.

Students enroll in two to six courses each quarter with four being a typical number. A course usually includes periods of instruction, but some courses are actually time periods reserved for meetings with Curricular Officers, seminars, special lectures, directed study or thesis research.

Each curriculum lasts from six to nine quarters. Consequently, approximately 200 new students enter NPS each quarter and 200 students graduate each quarter after studying at NPS for 1.5-2.5 years. [Ref.1]

B. BASIC PROBLEM DESCRIPTION

The academic course scheduling problem at NPS is described here in a general form. The following sections describe assumptions and simplifications that have been made, and the details of how the scheduling problem is modeled.

About a quarter in advance of the quarter being scheduled, students will have signed up for, or "requested" courses that departments are offering. Most of the courses requested will be dictated by the students' curricula, but there will be some electives. The schedulers takes these requests as fixed demands although, on rare occasions, the schedulers ask a student to change a request for an elective course.

The departments will already have a good idea of what the demand for courses will be *since the course sequences for* students is relatively inflexible. Therefore, the number of "segments" of each course (number of times the course is taught) , and which faculty members will teach the segments will already have been decided. The number of class periods, lab and/or lecture, for each segment is also predetermined. Therefore, the basic course scheduling problem is: Assign each segment with an accompanying faculty member to classrooms and laboratories so that the requisite number of class and lab periods is covered, and then assign students to those segments so that they can take all the courses they have requested. Students taking the same set of courses are grouped in to

"sections" and so assigning students is equivalent to assigning sections.

There are obvious feasiblity issues with the basic problem. We must ensure that (1) a faculty member teaches at most one course at any time (day and period), (2) sections are assigned to at most one course at any time, (3) a room has at most one course assigned to it at a time, (4) the number of students in a classroom at any time does not exceed the capacity of the room, and (5) classrooms and laboratories have the appropriate equipment for the lecture or laboratory part of the courses assigned to them.

There are also a few standard rules which should be satisfied: (1) Lectures of the same segment should be scheduled at the same time and in the same room. (2) Faculty members teaching 3 or more courses should teach at most two classes between 11 am and 1 pm to allow for a lunch break. (3) Similarly, sections should have at most 2 classes between 11 am and 1 pm. (4) Instructors' requests for "back-to-back" scheduling should be honored.

C. SIMPLIFYING ASSUMPTIONS

Based on the problem described above, we make certain assumptions to make the problem solvable:

1. Instructors have already been assigned to courses being offered and the number of segments to be assigned to an instructor is fixed.

2. A faculty member who teaches exactly two segments can request to have those courses scheduled "back-to-back" or has no preference in this regard. (This request is only a preference, which should, but may not be honored.) For simplicity's sake, we neglect faculty teaching three courses and assume they have no preference. (In the Glasgow Hall problem for the test quarter we considered, there is only one instructor out of sixty-six teaching three segments).

3. The number of students taking a particular course is known and dictates the number of segments of that course to be taught.

4. Every course is taught Monday through Thursday or Monday through Friday in the same room and at the same period. Laboratory sessions that might be scheduled in a different room or during a different period are ignored. This assumption essentially reduces the scheduling problem to scheduling on a single day.(In the Glasgow Hall data, only 7 of the 95 segments being taught do not satisfy the Monday-Thursday or Monday-Friday assumption.)

5. Because accelerated courses take up classrooms for the first half of the quarter and the classrooms they use are typically left free after the class is completed, they are treated like normal, full-quarter courses.

Given the assumptions and simplifications described above, the requirements of the NPS course scheduling problem are described below:

1. COURSE REQUIREMENTS:

- Every course segment is scheduled once per day in a given classroom.

2. CLASSROOM REQUIREMENTS:

- Classrooms can only be assigned to one course segment at the same period.

- To maintain a comfortable studying enviroment and to allow for students adding courses after the beginning of a quarter, the number of seats in a classroom must be 20% more than the number of students in a segment assigned to that room.

3. PERIOD REQUIREMENTS:

- Classes are taught in periods 1 through 7 since periods 8 and 9 are reserved for "special events".

- Students taking three or more courses should be allowed a lunch period between 11am-2pm.

- Faculty members teaching 3 courses should also be allowed a lunch period between 11am-2pm.

4. SECTION (STUDENT) REQUIREMENTS:

- Every section should be scheduled to take all the courses they have requested.

- A section can only be scheduled for one course in a particular period.

5. FACULTY REQUIREMENTS:

- A faculty member can only teach one class per period.

- Every faculty member must teach as many course segments as have been determined.

- A faculty member teaching exactly two segments may prefer to be scheduled "back-to-back".

Based on the preceding description, a prototypic integer programming model is presented in the next chapter to solve the NPS course scheduling problem.

III. MODEL

A mixed-integer programming model is developed to schedule NPS academic courses for a quarter. The objective is to minimize the deviation of the schedule from a "perfect" schedule, i.e., one that meets all of the instructors' and students' preferences and schedules each section for all of the courses it has requested.

A. INDICES AND SETS

The model has been formulated using five dimensions, or indices, on the various input parameters and decision variables. The indices and sets of indices are:

$f \in F$	faculty members (OR-WD, OR-RL, ..., etc.)
$s \in S$	sections (S001, S002, ..., etc.)
$c \in C$	courses (OA-4202, OA-4203, ..., etc.)
$r \in R$	classrooms (G109, G110, ..., etc.)
$p \in P$	periods (P1, P2, ..., P9)
$F^2 \subseteq F$	faculty member teaching exactly two segments
$F^3 \subseteq F$	faculty member teaching 3 or more segments
$F^B \subseteq F$	faculty members requesting back-to-back scheduling of courses
$S_c \subseteq S$	sections s requesting course c
$S^3 \subseteq S$	sections s requesting 3 or more courses
$C_f \subseteq C$	courses c taught by faculty f

$C_s \subseteq C$ courses c requested by section s

$P^L \subseteq P$ lunch time periods (P4, P5, P6)

B. DATA

The following list describes the various parameters required as input to the model:

$NumSeat_r$	number of seats in classroom r
$NumStudent_s$	number of students in section s
$NumSeg_{fc}$	number of segments of course c taught by faculty f
$RPenalty_r$	penalty for a less preferred classroom r (small value for large rooms on the first floor, increasing for smaller rooms and rooms not on the first floor)
$PPenalty_p$	penalty for a less preferred period p (small values for morning periods and large values for afternoon periods)
SPen1	per student penalty for allowing the number of students in a particular course to exceed the seating limit ($\lceil NumSeat_r /1.2 \rceil$) in classroom r
SPen2	penalty for allowing no lunch period for a section
SPen3	penalty for allowing no lunch period for a faculty member $f \in F^3$

SPen4 penalty for not scheduling a section to a requested course

SPen5 penalty for not scheduling back-to-back courses for a faculty member when requested

C. VARIABLES

1. DECISION VARIABLES

The decision variables in the model reflect the academic schedule for faculty members, sections and courses. They are described as follows:

X_{scrp} 1 if section s is scheduled for course c in classroom r in period p,and 0 otherwise

Y_{fcrp} 1 if faculty f is scheduled to teach course c in classroom r in period p, and 0 otherwise

2. ELASTIC VARIABLES

Elastic variables are necessary to allow penalized violation of constraints. For instance, it is better to not allow a faculty member a lunch period than to not schedule a section for a course. Therefore, the model must be able to violate at a penalty, a constraint which nominally requires a lunch period for a faculty member. But, at a higher penalty, the model will also allow a section to not be scheduled for a course since, in rare instances, it may be physically impossible.

$E1_{rp}$	number of students exceeding the room size limit for classroom r in period p
$E2_s$	1 if section $s \in S^3$ is not allowed a lunch period, and 0 otherwise
$E3_f$	1 if a faculty member $f \in F^3$ is not allowed a lunch period, and 0 otherwise
$E4_{sc}$	1 if section s is not scheduled for requested course c, and 0 otherwise
$E5_{fp}$	1 if faculty f member requesting back-to-back courses is scheduled for a course in period p but not scheduled in period p-1 or p+1, and 0 otherwise

D. MODEL FORMULATION

1. COURSE CONSTRAINTS

a. *NPerDay:* *Ensure that the correct number of segments is scheduled for each course.*

$$\sum_{r}\sum_{p} Y_{fcrp} = NumSeg_{fc} \quad \forall\, c \in C, f \in F$$

2. CLASSROOM CONSTRAINTS

a. *OnePerRoom:* Ensure that a classroom has at most one course assigned to it in any period.

$$\sum_{f \in F}\sum_{c \in C_f} Y_{fcrp} \leq 1 \quad \forall\, r \in R, p \in P$$

b. *RoomSize:* *Attempt to ensure that the number of seats in a classroom is at least 20% greater than the number of students in a segment assigned to that room.*

$$\sum_{s\in S_c} NumStudent_s * X_{scrp} \;-\; E1_{crp} \leq \left\lceil \frac{NumSeat_r}{1.2} \right\rceil$$

$$\forall \;\; c\in C,\; r\in R,\; p\in P$$

3. Period Constraints

a. *StdLunch:* *Students taking three or more courses should have a lunch hour between 11am and 2pm.*

$$\sum_{c\in C_s} \sum_{p\in P_{1t}} \sum_{r} X_{scrp} - E2_s \leq 2 \;\;\; \forall \; s\in S^3$$

c. *FctyLunch:* *Faculty members teaching three courses should have a lunch hour between 11am-2pm, also.*

$$\sum_{c\in C_f} \sum_{p\in P_{1t}} \sum_{r} X_{scrp} - E3_f \leq 2 \;\;\; \forall \;\; f\in F^3$$

4. LOGICAL CONSTRAINTS

a. *Logical:* *Ensure that a course is opened in a particular period and classroom before sections are assigned to it.*

$$X_{scpr} \;\leq\; Y_{fcrp} \qquad \forall \; r\in R, p\in P, s\in S, c\in C_S$$

5. SECTION CONSTRAINTS

a. *SameCourse:* *Every section should be scheduled for each course they have requested.*

$$\sum_{r} \sum_{p} X_{scrp} + E4_{sc} = 1 \quad \forall\ s \in S, c \in C_s$$

b. *OneCourse:* *A section can be scheduled to at most one course in each period.*

$$\sum_{c \in C_s} \sum_{r} X_{scrp} \le 1 \quad \forall\ s \in S, p \in P$$

6. FACULTY CONSTRAINTS

a. *OneLecture:* *A faculty member can be assigned to at most one lecture in each period.*

$$\sum_{c \in C_f} \sum_{r} Y_{fcrp} \le 1 \quad \forall\ f \in F, p \in P$$

c. BackToBack: Faculty members teaching exactly two segments should be scheduled back-to-back, if desired.

$$-\sum_{c \in C_f} \sum_{r} Y_{fcrp-1} + \sum_{c \in C_f} \sum_{r} Y_{fcrp} - \sum_{c \in C_f} \sum_{r} Y_{fcrp+1} - E5_{fp} \le 0$$

$$\forall\ f \in \{(F^B) \wedge (F^2)\}, p \in P$$

$$Y_{fcr0} \equiv Y_{fcr10} \equiv 0 \quad \forall\ f \in \{(F^B) \wedge (F^2)\}, r \in R$$

7. ***OBJECTIVE FUNCTION*** : Total penalty cost of the deviation of the schedule from a "perfect" schedule, i.e., one that meets all of the instructors' and students' preferences and schedules all sections to all courses requested.

$$MINIMIZE \sum_{s\in S_c} \sum_{c\in C_s} \sum_{r} \sum_{p} [RPenalty_r + PPenalty_p] * X_{scrp}$$

$$+ \sum_{s\in S_c} \sum_{c\in C_s} \sum_{r} \sum_{p} [RPenalty_r + PPenalty_p] * Y_{fcrp}$$

$$+ SPen1 * \sum_{c} \sum_{r} \sum_{p} E1_{crp} + SPen2 * \sum_{s} NumStudent_s * E2_s$$

$$+ SPen3 * \sum_{f} E3_f + SPen4 * \sum_{s\in S_c} \sum_{c\in C_s} NumStudent_s * E4_{sc}$$

$$+ SPen5 * \sum_{f\in F} \sum_{p} E5_{fp}$$

E. PARTIAL REFORMULATION

The LOGICAL constraints described in the previous section are called "variable upper bound" constraints (e.g.,[Ref.5], pg. 281) and make up the bulk of the constraints of the model. After converting each such constraint to an equality constraint by adding a surplus variable Z_{scrp} ,these constraints have the "consecutive 1s" property (e.g.,[Ref.5]) or more exactly, the "consecutive 1s and consecutive -1s" property. Consequently, they can be converted to network flow constraints.

The X-System, which will be used to solve the model can take advantage of this by "factorizing" these constraints which implies that an explicit inverse for these rows need not be maintained. (Preliminary tests on one problem showed that the size of the explicit inverse in the X-System dropped from about 4000x4000 to about 120x120; this is certainly worthwhile.) The LOGICAL constraints are converted to equality constraints as follows:

$$X_{scrp} + Z_{scrp} - Y_{fcrp} = 0 \quad \forall\ r\in R, p\in P, s\in S_c, c\in C_f, f\in F$$

$$Z_{scrp} \geq 0 \quad \forall\ r\in R, p\in P, s\in S, c\in C_S$$

Assume that the $s \in S_c$ are ordered $s_1, s_2, \ldots, s_{n(c)}$ where $n(c) = |S_c|$ and assume for fixed r,p and c, the constraints above are ordered i = 1, 2, ..., n(c). Then, by subtracting row (i) from row (i+1) for all rows i but the last, and by including the first row unchanged, we obtain the equivalent system:

$$X_{s_1crp} + Z_{s_1crp} - Y_{fcrp} = 0 \quad \forall\ r\epsilon R, p\epsilon P, c\epsilon C_f, f\epsilon F$$

$$-X_{s_{i-1}crp} - Z_{s_{i-1}crp} + X_{s_icrp} + Z_{s_icrp} = 0$$

$$\forall\ r\epsilon R, p\epsilon P, c\epsilon C_f, f\epsilon F\ i = 2, \ldots, n(c)$$

$$Z_{scrp} \geq 0 \quad \forall\ r\epsilon R, p\epsilon P, s\epsilon S, c\epsilon C_S$$

This system has exactly one +1 and one -1 in each column like a network constraint matrix should, except that Y_{fcrp} appears only once with a -1 coefficent. The X-System will factorize this partial network matrix so a redundant row need not be added, but we can also include the NPerDay constraints in the factorization and obtain a maximal network matrix in the variables X and Y.

IV. IMPLEMENTATION AND RESULTS

This chapter describes the implementation of the prototype NPS course scheduling model and provides computational results for a simplified problem. The creation of the data base for Glasgow Hall is described first.

A. THE DATABASE

The database has to hold all the necessary data, must be loadable entirely into memory and must be easily maintainable and accessible by GAMS. Therefore, the task is to set up appropriate data structures and fill them with selected data from registrar files (See Appendix A).

The procedures are described *as follows:*

1. From registrar files for Winter quarter 1993, we wrote a small SAS program (See Appendix B) to cluster individual students and their requested courses into a section-course file, i.e., students who take the same courses are clustered into a section (See Appendix C). Then, this data is imported into Lotus 1-2-3 to create two 123 "print files" (a file that can be directly imported into the GAMS model): a section-course file, i.e., the courses that each section has requested (See Appendix D), and section-number file, i.e., the number of students in each section (See Appendix E).

2. With the use of the master schedule for the 1993 Winter quarter, we create additional 123 print files --- a faculty-course file (courses that a faculty member must teach; see Appendix F), a number-segment file (number of segments per course; see Appendix G), and a faculty-preference file (instructors' preferences to be scheduled back-to-back or not; see Appendix H).

3. In our Glasgow Hall data, there are a total of 562 sections (1080 students), 81 courses and 66 faculty members. This covers about 55% of the sections, 31% of the faculty and 27% of the courses for the entire campus. A reduced data set for the Operations Research Department covers 19 faculty members, 83 sections and 26 courses.

4. Finally, for the sake of generality, we place the indices in a "set file" (Schedule.set) and the parameters(data) in a "data file" (Schedule.dat) from the main body of the GAMS model (See Appendices I and J).

These data are imported into the the GAMS model using the "$INCLUDE" statement. This arrangement allows the user the flexibility of running the model using different data sets without having to make any modifications to the main model.

B. IMPLEMENTATION

The model described in the previous chapter was implemented in the Generalized Algebraic Modeling System (GAMS) [Ref.3]. The model is implemented very closely to the mathematical formulation except that the index f is dropped from the variable Y_{fcrp}. This is possible because, for this data, each course is taught by a unique faculty member (See Appendix K). An attempt to solve the Glasgow Hall problem for Winter 1992 failed. This problem has 66 faculty members, 562 sections, 104 courses and 17 classrooms and results in a model, which was successfully generated but not solved, having 148,161 constraints and 287,778 variables. This data set was therefore reduced to encompass just the OR Department which covers 19 faculty members, 83 sections and 26 courses and results in a model with less than 17,000 constraints and 32,000 variables.

The solver used in testing is the X-System [Ref.4] which is a primal/dual linear programming solver with integer and nonlinear capabilities The X-System represents elastic variables in a semi-implicit manner which makes implementation of elastic constraints particularly efficient. Initial tests showed that the X-System solver could solve the linear programming (LP) relaxation of the OR problem but that numerical instabilities caused failure after the solver went into the branch-and-bound, integer programming part of the

code. (Experiments with several other solvers showed they could not even solve the LP relaxation.) Therefore, two modest simplifications to the model and data were made which resulted in a solveable problem. In particular, the "BackToBack" constraints were deleted since this would annoy at most 3 faculty members (out of 19) who expressed a preference to have their courses scheduled back to back. Also, each room capacity was increased to the actual number of students that will fit into the room, $NumSeat_r$.

The OR scheduling model was solved on an Amdahl 5995-700A mainframe computer using 256 megabytes of memory. The model has 16,649 constraints and 31,381 variables, was generated in 229.6 seconds and solved to within 0.2% of optimality in 3258.8 seconds. The solution is given in Appendix L.

The schedule was validated by the NPS Registrar [Ref. 6] and found to be quite reasonable. No elastic variables were positive so all students could take their requested courses, all faculty and students were allowed a period for lunch, and (relaxed) room capacity constraints were satisfied.

V. CONCLUSIONS AND RECOMMENDATIONS

This thesis has developed a prototypic integer programming model for the quarterly scheduling of academic courses at the Naval Postgraduate School. The size of the problem was reduced significantly over the size necessary to schedule all courses at the school and some simplifying assumptions were made. However, the method shows promise and should be explored further.

Course scheduling at the Naval Postgraduate School is dissimilar to course scheduling at civilian universities in that students must (almost always) be scheduled for the courses they request from the set of courses being taught during the quarter. An integer programming model to solve the scheduling problem is, consequently, quite complicated. It must include variables and constraints to handle the assignment of students to courses, periods and classrooms, in addition to variables and constraints to handle the assignment of faculty members to courses in particular periods and classrooms.

The model developed in this thesis was implemented in the Generalized Algebraic Modeling System and generated for a reduced data set including the Operations Research, Mathematics and National Security Affairs courses which are

primarily taught in a single building, Glasgow Hall. The resulting model contained 287,778 variables and 148,161 constraints and was too large to solve. Consequently, the data was further simplified to encompass only the Operations Research Department. The resulting model had less than 32,000 variables and 17,000 constraints and was solved using the X-System on an Amdahl 5995-700A, requiring a total of 3488.4 seconds. The solution satisfied all scheduling requirements except for back-to-back scheduling preferences for three faculty members. The solution was also validated with the NPS Registrar and found to be reasonable.

The model, with modest generalizations, may be useful for a single department in suggesting a schedule for the department to the school's scheduling staff. However, further research and other mathematical formulations may lead to solutions which are directly, or almost directly implementable for a sizeable segment of the school such as for the courses taught in Glasgow Hall. For instance, a reformulation of the model did convert "variable upper bound" constraints into network constraints which speeded up solutions dramatically. Further reformulations may make even larger problems solveable.

In addition to reformulating the model for ease of solution, the model must be extended to incorporate a number of scheduling requirements that were ignored for simplicity's sake. The main simplifying assumption made was that each

course is taught Monday through Thursday, or Friday, in the same room and at the same period. This essentially reduces the scheduling problem to scheduling on a single day. This is reasonable for about 90% of the classes which are taught in the same room Monday through Thursday, or Monday through Friday, or Monday, Wednesday, Friday, etc. This is true because course lectures are always scheduled for the same room and period, and courses taught on the days mentioned above conflict under this restriction. However, the model must be extended to handle courses that have lectures on several days at the same period and in the same room, but have laboratory sessions in different rooms and possibly different periods. This will require significant work.

A number of simplifications were made whose generalization will require only a modest amount of data manipulation and may actually simplify solution of the model. For instance, "black-out" periods for faculty were ignored. Such periods might indicate that a faculty member has weekly administration meetings for a particular set of periods and simply cannot be scheduled to teach a class during those periods. Excluding scheduling possibilities for such a faculty member will eliminate a number of variables in the model. If faculty have "black-out preferences", periods when they would prefer not to teach, these could be handled by just putting a large cost on any variables that would assign that faculty member to a course during an undesireable period. Provisions will also

have to be added to the model to allow fixing of variables. For instance, it will sometimes be expedient for schedulers to fix a course segment for a faculty member to a particular period and room even though, technically, other rooms and periods might be acceptable. This could be accomplished by fixing the appropriate Y_{fcrp} to 1, which is not hard to do in GAMS and would probably speed up solution of the model.

Further work on the expansion of the model and the data base used for the model is necessary to reach a useful and reliable scheduling system. This will enable individual treatment of requests, quick response to changes in the system, decrease the number of enrollment changes and produce a high quality schedule.

APPENDIX A. REGISTRAR FILES

NM	CLSCD
.	
.	
.	
ADAMS, COLWELL	MA11172
ADAMS, DEIST J.	CC30012
ADAMS, DEIST J.	CS29701
ADAMS, DEIST J.	MA12484
ADAMS, DEIST J.	OS21032
ADAMS, MORRIS	OA0810
ADAMS, MORRIS	OA3105
ADAMS, MORRIS	OA3900
ADAMS, MORRIS	OA4102
ADAMS, MORRIS	OA4303
.	
.	
.	
ZANGARDI, JORDAN	MA23003
ZANGARDI, JORDAN	MN21503
ZANGARDI, JORDAN	MN31722
ZANGARDI, JORDAN	MN33331
.	
.	
.	

This file lists all of the students who request at least one OR, Math or NSA course.

Example: ADAMS, MORRIS request OA0810, OA3105, OA3900, OA4102 and OA4303.

(For the sake of privacy, all names list above are imaginary.)

APPENDIX B.
<u>SAS PROGRAM</u>

```
DATA ONE;
INFILE 'WU DATA *';
INPUT
   NAME     $  1-28
   CLASS    $ 31-40;
N=1;
PROC SORT;
   BY NAME CLASS;
PROC MEANS NOPRINT;
   BY NAME;
      OUTPUT OUT=COUNTS N=N;
DATA TWO;
   SET COUNTS ONE;
PROC SORT OUT=THREE;
   BY NAME;
DATA FOUR;
   SET THREE;
FILE 'NEWWU DATA T';
   PUT
      NAME $ 1-30
      CLASS $ 31-40
      N      45;
DATA TWO;
   LENGTH CLASSES $ 80;
   INFILE 'NEWWU DATA T';
INPUT
   NAME $ 1-30
   N      45;
ARRAY CLASS(*) $ CLASS1 CLASS2 CLASS3 CLASS4 CLASS5
                 CLASS6 CLASS7 CLASS8 CLASS9;
   DO COUNT = 1 TO N;
      INPUT
      CLASS(COUNT) $ 31 - 40;
   END;
CLASSES=
COMPRESS(CLASS1||'*'||CLASS2||'*'||CLASS3||'*'||CLASS4||'*'
        ||CLASS5||'*'||CLASS6||'*'||CLASS7||'*'||CLASS8||'*'
        ||CLASS9);
PROC SORT;
   BY N CLASSES;
PROC PRINT;
   BY N CLASSES;
   VAR NAME;
```

APPENDIX C.

CLUSTER STUDENTS INTO SECTIONS

```
          .
          .
          .

--------------- N=4
*****CLASSES=MA31103*NS3000*OA31022*OA32001*****
               OBS         NAME
               551    VANKAN, MORRIS D.

--------------- N=4
*****CLASSES=MA31103*OA31022*OA32001*OA3610*****
               OBS         NAME
               552    DAVIS, JONE E.
               553    HALVORSON, TYLER B.
               554    PENNYPACKER, BRUNO S.
               555    REDMAN, TARSA
               556    ROBERTS, WIMER L.
               557    SPERRY, DOWNS
               558    WILSON, MARK

--------------- N=4
*****CLASSES=MA31321*ME2201*ME2601*MS2201*****
               OBS         NAME
               560    AVENGER, DOYLE
               561    PALL, PAXTON E.
               562    SUAREZ, COOKE
          .
          .
          .
```

This file shows students requesting the same courses have been grouped into section.

Example: DOYLE AVENGER, PAXTON E. PALL and COOKE SUARZE have requested the courses MA31321, ME2201, ME2601 and MS2201.

(For the sake of privacy, all names above are imaginary.)

APPENDIX D.

SECTION-COURSE (SCourse.PRN)

```
     .
     .
     .
S046.(OA3105,OA4301,OA4302)
S047.(OA3601,OA3602,OA4301)
S048.(OA3601,OA4301,OA4302)
S049.(OA3601,OA4301,OA4612)
S050.(OA3602,OA4102,OA4303)
S051.(OA4102,OA4301)
S052.(OA4102,OA4301)
S053.(OA4203,OA4301,OA4302)
S054.(OA4301,OA4602,OA4303)
S055.(OA4301,OA4302)
S056.(OA4301,OA4654,OA4655)
S057.(OA4301,OA4654,OA4655)
S058.(OA3302,OA4202)
S059.(OA3104,OA3302,OA3601,OA4202)
S060.(OA3104,OA3302,OA4202,OA4501)
S061.(OA3104,OA3302,OA4202,OA4604)
S062.(OA3104,OA3302,OA4202,OA4910)
S063.(OA3105,OA4301,OA4302)
S064.(OA3601,OA3602,OA4204,OA4301)
S065.(OA3601,OA4204,OA4301)
S066.(OA4301,OA4654,OA4655,OA4910)
S067.(OA4301,OA4654,OA4655)
S068.OA4602
S069.OA3104
S070.(OA3102,OA3200)
S071.OA3102
S072.(OA3105,OA3900,OA4602)
S073.(OA3105,OA3900,OA4102)
S074.(OA3900,OA4654,OA4655)
S075.(OA3105,OA3900,OA4102,OA4303)
     .
     .
     .
```

This file shows the secton number and the courses the section has requested.

Example: Section #075 (S075) has requested OA3105, OA3900, OA4102 and OA4303.

APPENDIX E.

NUMBER OF STUDENTS PER SECTION (SNUMBER.PRN)

	NumStud
.	
.	
.	
S046	1
S047	2
S048	1
S049	6
S050	1
S051	1
S052	2
S053	4
S054	3
S055	1
S056	3
S057	3
S058	2
S059	1
S060	4
S061	11
S062	3
S063	1
S064	1
S065	3
S066	3
S067	1
S068	1
S069	1
S070	1
S071	1
.	
.	
.	

This file shows the number of students in each section.

Example: Section #061 (S061) has 11 students in it.

APPENDIX F.

FACULTY-COURSE (FCourse.PRN)

```
.
.
.
OR-BA.OA3302
OR-CO.(OA3601,OA4605)
OR-DE.OA3200
OR-ER.OA3602
OR-EY.OA4302
OR-GV.OA4910
OR-HL.OA4602
OR-HT.OA3610
OR-LA.(OA3104,OA3105)
OR-LS.OA4303
OR-MD.OA4604
OR-MT.OA4301
OR-PY.(OA4654,OA4655)
OR-RE.OA3102
OR-SO.OA4612
OR-WD.(OA4202,OA4203)
OR-WS.OA4204
.
.
.
```

This file shows the courses that a faculty member must teach.

Example: Professor Read in the OR Department (OR-RE) will teach OA3102.

APPENDIX G.

FACULTY PREFERENCE (FPrefer.PRN)

	PREFER
.	
.	
.	
OR-LS	
OR-MD	
OR-MH	
OR-MJ	
OR-MT	1
OR-MY	
OR-PY	
OR-RE	1
OR-RO	
OR-SO	
OR-SY	
OR-TH	
OR-TW	
OR-WD	
OR-WO	
OR-WS	
.	
.	
.	

Example: Professor Marshall in the OR department (OR-MT) requests that his courses be scheduled back-to-back while Professor Woods (OR-WO) does not care.

APPENDIX H.

NUMBER SEGMENT PER COURSE (NUMSEG.PRN)

```
     .
     .
     .
NUMSEG("OR-CO","OA3601")=1;
NUMSEG("OR-CO","OA4605")=1;
NUMSEG("OR-DE","OA3200")=2;
NUMSEG("OR-ER","OA3602")=1;
NUMSEG("OR-EY","OA4302")=1;
NUMSEG("OR-GV","OA4910")=1;
NUMSEG("OR-HL","OA4602")=1;
NUMSEG("OR-HT","OA3610")=1;
NUMSEG("OR-LA","OA3104")=1;
NUMSEG("OR-LA","OA3105")=1;
NUMSEG("OR-LS","OA4303")=1;
NUMSEG("OR-MD","OA4604")=1;
NUMSEG("OR-MT","OA4301")=2;
NUMSEG("OR-PY","OA4655")=1;
NUMSEG("OR-RE","OA3102")=3;
NUMSEG("OR-SO","OA4612")=1;
NUMSEG("OR-WD","OA4202")=1;
NUMSEG("OR-WD","OA4203")=1;
NUMSEG("OR-WS","OA4204")=1;
     .
     .
     .
```

This file shows the number of segments a faculty member will teach for a particular course.

Example: Professor Marshall in the OR department (OR-MT) will teach 2 segments of course OA4301.

APPENDIX I.

INDICES OF MODEL (SCHEDULE.SET)

```
SETS  F     faculty members
      /  AS-MR
         CC-FU
         MA-FF, MA-FI, MA-FR, MA-GR, MA-HE, MA-HT, MA-HV, MA-LE
         MA-MA, MA-ND, MA-ON, MA-RA, MA-RU, MA-SA, MA-TH, MA-WC, MA-ZH
         NS-AH, NS-CH, NS-EY, NS-GT, NS-LB, NS-LX, NS-MI, NS-MK, NS-OS
         NS-PA, NS-PR, NS-RB, NS-SC, NS-SK, NS-TK, NS-TO, NS-TR, NS-TT
         NS-WB, NS-WZ
         OR-BA, OR-BR, OR-CO, OR-DE, OR-ER, OR-EY, OR-GV, OR-HL, OR-HT
         OR-LA, OR-LS, OR-MD, OR-MH, OR-MJ, OR-MT, OR-MY, OR-PY, OR-RE
         OR-RO, OR-SO, OR-SY, OR-TH, OR-TW, OR-WD, OR-WO, OR-WS, OR-ZI
         /

      S     sections
      /  S001*S562  /

      C     courses
      /  MA0117, MA0118, MA0125, MA0142, MA1042, MA1117, MA1118, MA1248
         MA1248, MA1248, MA2049, MA2121, MA2300, MA3026, MA3110, MA3132
         MA3139, MA3232, MA4027, MA4323
         NS3000, NS3011, NS3012, NS3023, NS3037, NS3041, NS3159, NS3230
         NS3252, NS3300, NS0331, NS3320, NS3400, NS3410, NS3460, NS3520
         NS3663, NS3880, NS3881, NS4152, NS4200, NS4250, NS4300, NS4410
         NS4510, NS4660, NS4710
         OA0200, OA2900, OA3102, OA3104, OA3105, OA3200, OA3302, OA3601
         OA3602, OA3610, OA3900, OA4102, OA4202, OA4203, OA4204, OA4301
         OA4302, OA4303, OA4501, OA4602, OA4604, OA4605, OA4612, OA4654
         OA4655, OA4910
         OS2103, OS3004, OS3006,  OS3008, OS3302, OS3404, OS3601, OS3602
         OS3604, OS4701  /

      R     classrooms
      /  G109, G110, G113, G114, G115, G117, G118, G122, G129, G130, G133
         GB13, GB14, GB15,  GB17, GB18, GB19  /

      P     periods
      /  P1, P2, P3, P4, P5, P6, P7/

      ;

SET   LunchTime(P)   periods are reserved for lunch time
      /  P4, P5, P6  /
      ;
```

APPENDIX J.

DATA SET OF MODEL (SCHEDULE.DAT)

```
PARAMETER NumSeat(R)      number of seats per classroom
  /
  G109      180, G110     36, G113      36, G114     36, G115      40
  G117       27, G118     36, G122      44, G129     36, G130      36
  G133      36
  GB13      36, GB14      32, GB15      28, GB17     28, GB18      32
  GB19      36
  /   ;

PARAMETER RPenalty(R)     penalty for scheduling in a "bad" classroom
  /
  G109        8, G110      2, G113       2, G114      2, G115       1
  G117        3, G118      2, G122       1, G129      2, G130       2
  G133        2
  GB13        4, GB14      5, GB15       6, GB17      6, GB18       5
  GB19        4
  /   ;

PARAMETER PPenalty(P)     penalty for scheduling in a "bad" period
   /
   P1       16, P2        8, P3         8, P4        16, P5        24
   P6       40, P7       48
   /  ;
```

APPENDIX K.

GAMS MODEL

```
$TITLE   NAVAL POSTGRADUADE SCHOOL SCHEDULING SYSTEM
$offupper offsymxref offsymlist offuellist
$ontext
      By:         Chinese Army Major  Wu, Hsi-Hsien
      Advisor:    Professor  R. Kevin Wood
                  Professor  Richard E. Rosenthal
      Description:

                This prototypic integer programming model is developed to
               aid in solving the NPS academic course scheduling program.
$offtext
options
  LIMROW     =     0
  SOLPRINT   =    OFF
  LIMCOL     =     0
* MIP        =    OSL
* MIP        =     XA
  MIP        =     XS
  ITERLIM    = 900000
  RESLIM     =  66000
  ;
SETS
   F   faculty members
   S   sections
   C   courses
   R   classrooms
   P   periods
   ;
$INCLUDE SCHEDULE SET
SET LunchTime(P)     periods are reserved for lunch time;

ALIAS (C,C1) ;
ALIAS (R,RR) ;
ALIAS (P,PA) ;
ALIAS (S,SP) ;

PARAMETERS
*data of course and classroom :
   NumSeat(R)      number of seats per classroom
*data of penalty cost :
   RPenalty(R)     penalty for scheduling in a "bad" classroom
   PPenalty(P)     penalty for scheduling in a "bad" period
   ;

$INCLUDE SCHEDULE DATA
*data listed below comes from LOTUS-123 spreadsheet :
*---data for faculty members---
SET FCTYCOURSE(F,C)  COURSES C TAUGHT BY FACULTY MEMBER F
$INCLUDE 'FCOURSE PRN'
;
TABLE FPREFER(F,*)   FACULTY MEMBER PREFER BLACK-OUT OR
```

BACK-TO-BACK

```
$INCLUDE 'FPREFER PRN'
;
PARAMETER FNUMCOURSE(F)   NUMBER OF COURSE THAT A FACULTY MUST
                          TEACH;
   PARAMETER FctyPrefer(F) the preference of a faculty member;
   FctyPrefer(F)=FPrefer(F,"PREFER")
      ;
PARAMETER NUMSEG(F,C)   NUMBER OF SEGMENTS THAT A FACULTY MUST
                        TEACH;
$INCLUDE 'FNUMSEG PRN'
   PARAMETER TNUMSEG(F) TOTAL NUMBER OF SEGMENTS OF FACULTY F
      TNUMSEG(F)=SUM(C,FNUMSEG(F,C));

*---data for sections---
SET SECCOURSE(S,C)   COURSES C WHICH IS TAKEN BY SECTION S
$INCLUDE 'SCOURS PRN'
;
PARAMETER TEMP, SECC(S,C);
LOOP(C,
   TEMP=0;
   LOOP (S$SECCOURSE(S,C),
        TEMP=TEMP+1;
        SECC(S,C)=TEMP;
   );
);
DISPLAY SECC;
TABLE SecNumber(S,*)   number of students per sections
$INCLUDE 'SNUMBER PRN'
;
PARAMETER NumStudent(S)   number of students per section;
   NumStudent(S)=SecNumber(S,"NumStud");

SCALARS
   SPenalty   penalty cost for not schedul a section
   ;
   SPENALTY=SMAX( R, RPENALTY(R))+SMAX( P, PPENALTY(P)) ;

 POSITIVE VARIABLES
   Z(S,C,R,P)     SURPLUS FOR VARIABLE UPPER BOUND CONSTRAINTS
   E5(R,P)        Room too small
   E6(S)          No student lunch
   E7(F)          No faculty lunch
   E13(S,C)       Section not scheduled
   E18(F,P)       Cannot get back to back classes
   ;

BINARY VARIABLES
   X(S,C,R,P)    YES-OR-NO FOR SCHEDULE SECTION S COURSE C AT
                 CLASSROOM R PERIOD P AND WEEKDAY W
   Y(C,R,P)      YES-OR-NO FOR SCHEDULE COURSE C AT CLASSROOM
                 R PERIOD P and weekday W
   ;

FREE VARIABLE
   OBJ            total penalty cost
   ;

EQUATIONS

*logical constraints   :
```

```
   LOGICAL(R,P,C,S)      SECTIONS SCHEDULED TO OPEN COURSES ONLY


*course constraints      :
   NPERDAY(F,C)           ensure that the correct number of segments is
                          scheduled for each course

*classroom constraints :
   ONEPERROOM(R,P)        at most one course per room in any
                          period
   ROOMSIZE(C,R,P)        omsize has at least 20% extra capacity

*period constraints      :
   STDLUNCH(S)           STUDENTS GET ONE LUNCH HOUR BETWEEN
                          11am-2pm
   FCTYLUNCH(F)          FACULTY WITH 3 COURSES GET a lunch hour
                          in 11am-2pm


*section constraints     :
   SAMECOURSE(S,C)       EVERY SECTION SHOULD BE SCHEDULED
   ONECOURSE(S,P)        At most one course for a section in any
                          period

*faculty constraints     :
   ONELECTURE(F,P)      At most one lecture per faculty member
                          in a period
   BACKTBACK(F,C,R,P) Faculty prefer back-to-back if teaching
                          2 courses

*---objective function---
   OBJDEF         define objective function--total penalty cost
   ;

LOGICAL(R,P,C,S) $ (SECCOURSE(S,C)) ..
           X(S,C,R,P) + Z(S,C,R,P)
 - SUM(SP$(SECC(S,C) NE 1 AND SECC(SP,C) EQ SECC(S,C)-1),
           X(SP,C,R,P) + Z(SP,C,R,P) )
 - Y(C,R,P) $ (SECC(S,C) EQ 1) =E= 0;

NPERDAY(F,C)$FCTYCOURSE(F,C)..
     SUM( (R,P), Y(C,R,P) ) =E= NUMSEG(F,C) ;

ONEPERROOM(R,P) ..
     SUM( C, Y(C,R,P) ) =L= 1 ;

ROOMSIZE(C,R,P) ..
      SUM( S $ (SECCOURSE(S,C) ),
      NUMSTUDENT(S) * X(S,C,R,P) ) - E5(C,R,P)
     =L= CEIL(NUMSEAT(R)/1.2);

STDLUNCH(S)..
     SUM( (C,R,P) $  (SECCOURSE(S,C) ),
     X(S,C,R,P) $ LUNCHTIME(P) ) -E6(S) =L= 2 ;

FCTYLUNCH(F)..
     SUM( (C,R,P) $  (FCTYCOURSE(F,C)) ,
     Y(C,R,P) $ LUNCHTIME(P) ) -E7(F) =L= 2 ;

SAMECOURSE(S,C) $ (SECCOURSE(S,C) )..
     SUM( (R,P) ,  X(S,C,R,P) )  +E13(S,C) =E= 1 ;
```

```
ONECOURSE(S,P) ..
     SUM( (C,R) $ (SECCOURSE(S,C)),  X(S,C,R,P) ) =L= 1 ;

ONELECTURE(F,P) ..
     SUM( (C,R) $ (FCTYCOURSE(F,C)), Y(C,R,P) ) =L= 1 ;

BACKTBACK(F,C,R,P) $ (FCTYCOURSE(F,C)$((FCTYPREFER(F) EQ 0)
                       AND (TNUMSEG(F) EQ 2)))..
SUM( (C1,RR,PA) $(FCTYCOURSE(F,C1)$((((ORD(C))-(ORD(C1)))NE 0)
                       AND (ABS((ORD(P))-(ORD(PA))) EQ 1))),
      Y(C1,RR,PA) ) + E18(F,C,R,P) =E= 1  ;

OBJDEF..
     0.001 * ( SUM( (S,C,R,P) $ (SECCOURSE(S,C) ) ,
       (RPENALTY(R) + PPENALTY(P) ) * X(S,C,R,P)
     + (RPENALTY(R) + PPENALTY(P) ) * Y(C,R,P) )
     + 1.5 * SPENALTY * SUM((C,R,P), E5(C,R,P) )
     + 1.1 * SPENALTY * SUM(S, NUMSTUDENT(S)*E6(S))
     + 1.2 * SPENALTY * SUM(F, E7(F))
     + 2 * SPENALTY *(SUM((S,C)$SECCOURSE(S,C)
                         NUMSTUDENT(S)* E13(S,C) )
     + SPENALTY * SUM((F,C,R,P) $ (FCTYCOURSE(F,C)),
        E18(F,C,R,P)  ) )   =E= OBJ ;

MODEL SCHEDULE /ALL/ ;
*..........................X-SYSTEM OPTIONS
 SCALAR SOLVERXS;
 SOLVERXS = 1;
 FILE OPTFIL /XS OPT A/;
 IF( SOLVERXS EQ 1,
     OPTION LP   = XS,
            RMIP = XS,
            MIP  = XS;
     SCHEDULE.OPTFILE = 1;
     PUT OPTFIL;
     PUT "*(BASIC LP OPTIONS)                          "/;
     PUT " XREF                                        "/;
     PUT " ELASTIC E5                                  "/;
     PUT " ELASTIC E6                                  "/;
     PUT " ELASTIC E7                                  "/;
     PUT " ELASTIC E13                                 "/;
     PUT " ELASTIC E18                                 "/;
     PUT " FACTOR LOGICAL                              "/;
     PUT " FACTOR NPERDAY                              "/;
     PUT " PURE_NET                                    "/;
     PUT " PRIMAL                                      "/;
     PUT " KXD          77000                          "/;
     PUT " KBR          -1                             "/;
     PUT " PRINT        1                              "/;
     PUT " MAX_MINUTES    60                           "/;
     PUT "*(BASIC MIP OPTIONS)                         "/;
     PUT " MAX_DEPTH     90                            "/;
     PUT "*OVERRIDE GAMS TOLERANCE OPTCR               "/;
     PUT " OPTCR         0.10                          "/;
     PUT " MAX_BACK      400                           "/;
     PUT " MAX_NODES     400                           "/;
     PUT " MXKRS         0                             "/;
     PUT "*(OTHER BASIC OPTIONS)                       "/;
     PUT " *RMIP                                       "/;
     PUT " PRE-REDUCE                                  "/;
     PUT "*END OF OPTION FILE XS.OPT                   " ;
```

```
     PUTCLOSE OPTFIL;
   );
SOLVE SCHEDULE USING MIP MINIMIZING OBJ ;
DISPLAY X.L, Y.L, E5.L, E6.L, E7.L, E13.L, E18.L;
```

APPENDIX L.

LISTING OF RESULTS FOR THE OR DEPARTMENT

A. FACULTY SCHEDULE:

Faculty	Course	Classroom	Period						
			P1	P2	P3	P4	P5	P6	P7
OR-RE	OA-3102	G115						1	
		G115							1
		G122				1			
OR-LA	OA-3104	G114							1
	OA-3105	G110				1			
OR-DE	OA-3200	G122						1	
		G129							1
OR-BA	OA-3302	G115		1					
OR-CO	OA-3601	G115				1			
	OA-4605	G110						1	
OR-ER	OA-3602	G110				1			
OR-HT	OA-3610	G122					1		
OR-ZI	OA-3900	G122	1						
OR-LS	OA-4102	G115			1				
	OA-4303	G115					1		
OR-WD	OA-4202	G129	1						
	OA-4203	G114						1	
OR-WS	OA-4204	G113			1				

OR-MT	OA-4301	G115	1						
		G122							1
OR-EY	OA-4302	G122			1				
AS-MR	OA-4501	G114			1				
OR-HL	OA-4602	G114		1					
OR-MD	OA-4604	G130				1			
OR-SO	OA-4612	G110		1					
OR-PY	OA-4654	G110			1				
	OA-4655	G122		1					
OR-GV	OA-4910	G133						1	

B. SECTION SCHEDULE:

Section	Course	Classroom	Period						
			P1	P2	P3	P4	P5	P6	P7
S001	OA-3302	G115		1					
S002	OA-3102	G115							1
	OA-3610	G122					1		
S003	OA-4910	G133						1	
S004	OA-3102	G115						1	
S005	OA-3102	G115							1
S006	OA-3102	G115							1
	OA-3200	G122						1	
S007	OA-3102	G115							1
	OA-3200	G122						1	
S008	OA-3200	G122						1	

S009	OA-3102	G115							1
	OA-3200	G122						1	
S010	OA-3610	G122					1		
S011	OA-3610	G122					1		
S012	OA-3610	G122					1		
S013	OA-4654	G110			1				
	OA-4655	G122		1					
S014	OA-3104	G114							1
	OA-3302	G115		1					
	OA-4202	G129	1						
S015	OA-3601	G115				1			
	OA-4301	G122							1
	OA-4612	G110		1					
S016	OA-4301	G115	1						
	OA-4654	G110			1				
	OA-4655	G122		1					
S017	OA-4602	G114		1					
S018	OA-4605	G110						1	
S019	OA-4654	G110			1				
	OA-4655	G122		1					
S020	OA-4654	G110			1				
	OA-4655	G122		1					
S021	OA-3102	G122				1			
	OA-3200	G122						1	
S022	OA-3302	G115		1					
S023	OA-3102	G122				1			
	OA-3200	G122						1	
S024	OA4203	G114						1	
S025	OA-4102	G115			1				
S026	OA-3102	G115						1	

S027	OA-3102	G115						1	
S028	OA-3102	G115							1
S029	OA-3102	G115							1
S030	OA-3102	G115							1
	OA-3200	G122						1	
S031	OA-3102	G115							1
	OA-3200	G122						1	
S032	OA-3102	G122				1			
	OA-3200	G122						1	
S033	OA-3102	G115						1	
	OA-3200	G129							1
S034	OA-3102	G122				1			
	OA-3200	G122						1	
S035	OA-3102	G115							1
	OA-3200	G122						1	
S036	OA-3102	G115							1
	OA-3200	G122						1	
	OA-3610	G122					1		
S037	OA-4301	G115	1						
S038	OA-3104	G114							1
	OA-3302	G115		1					
	OA-4202	G129	1						
S039	OA-3610	G122					1		
S040	OA-4102	G115			1				
	OA-4301	G122							1
S041	OA-4203	G114						1	
	OA-4301	G122							1
	OA-4302	G122			1				
S042	OA-3900	G122	1						
	OA-4604	G130				1			

S043	OA-4203	G114						1	
	OA-4301	G115	1						
	OA-4302	G122			1				
S044	OA-3105	G110				1			
	OA-3602	G110							1
	OA-4303	G115					1		
S045	OA-3105	G110				1			
	OA-4102	G115			1				
	OA-4301	G122							1
S046	OA-3105	G110				1			
	OA-4301	G122							1
	OA-4302	G122			1				
S047	OA-3601	G115				1			
	OA-3602	G110							1
	OA-4301	G115	1						
S048	OA-3601	G115				1			
	OA-4301	G115	1						
	OA-4302	G122			1				
S049	OA-3601	G115				1			
	OA-4301	G115	1						
	OA-4612	G110		1					
S050	OA-3602	G110							1
	OA-4102	G115			1				
	OA-4303	G115					1		
S051	OA-4102	G115			1				
	OA-4301	G115	1						
S052	OA-4102	G115			1				
	OA-4301	G115	1						
S053	OA-4203	G114						1	
	OA-4301	G122							1

	OA-4302	G122			1				
S054	OA-4301	G122							1
	OA-4303	G115					1		
	OA-4602	G114		1					
S055	OA-4301	G115	1						
	OA-4302	G122			1				
S056	OA-4301	G115	1						
	OA-4654	G110			1				
	OA-4655	G122		1					
S057	OA-4301	G115	1						
	OA-4654	G110			1				
	OA-4655	G122		1					
S058	OA-3302	G115		1					
	OA-4202	G129	1						
S059	OA-3104	G114							1
	OA-3302	G115		1					
	OA-3601	G115				1			
	OA-4202	G129	1						
S060	OA-3104	G114							1
	OA-3302	G115		1					
	OA-4202	G129	1						
	OA-4501	G114			1				
S061	OA-3104	G114							1
	OA-3302	G115		1					
	OA-4202	G129	1						
	OA-4604	G130				1			
S062	OA-3104	G114							1
	OA-3302	G115		1					
	OA-4202	G129	1						
	OA-4910	G133						1	

S063	OA-3105	G110				1			
	OA-4301	G115	1						
	OA-4302	G122			1				
S064	OA-3601	G115				1			
	OA-3602	G110							1
	OA-4204	G113			1				
	OA-4301	G115	1						
S065	OA-3601	G115				1			
	OA-4204	G113			1				
	OA-4301	G115	1						
S066	OA-4301	G122							1
	OA-4654	G110			1				
	OA-4655	G122		1					
	OA-4910	G133						1	
S067	OA-4301	G115	1						
	OA-4654	G110			1				
	OA-4655	G122		1					
S068	OA-4602	G114		1					
S069	OA-3104	G114							1
S070	OA-3102	G115							1
	OA-3200	G122						1	
S071	OA-3102	G115						1	
S072	OA-3105	G110				1			
	OA-3900	G122	1						
	OA-4602	G114		1					
S073	OA-3105	G110				1			
	OA-3900	G122	1						
	OA-4102	G115			1				
S074	OA-3900	G122	1						
	OA-4654	G110			1				

	OA-4655	G122		1					
S075	OA-3105	G110				1			
	OA-3900	G122	1						
	OA-4102	G115			1				
	OA-4303	G115					1		
S076	OA-3601	G115				1			
	OA-3602	G110							1
	OA-4301	G115	1						
	OA-4302	G122			1				
S077	OA-3601	G115				1			
	OA-4102	G115			1				
	OA-4301	G115	1						
	OA-4612	G110		1					
S078	OA-3601	G115				1			
	OA-4301	G122							1
	OA-4302	G122			1				
	OA-4910	G133						1	
S079	OA-3602	G110							1
	OA-3900	G122	1						
	OA-4602	G114		1					
	OA-4605	G110						1	
S080	OA-3602	G110							1
	OA-4301	G115	1						
	OA-4302	G122			1				
	OA-4303	G115					1		
S081	OA-3900	G122	1						
	OA-4204	G113			1				
	OA-4602	G114		1					
	OA-4605	G110						1	
S082	OA-3900	G122	1						

	OA-4302	G122			1				
	OA-4602	G114		1					
	OA-4605	G110						1	
S083	OA-3900	G122	1						
	OA-4303	G115					1		
	OA-4602	G114		1					
	OA-4605	G110						1	

C. CLASSROOM SCHEDULE:

Class-room	Period						
	P1	**P2**	**P3**	**P4**	**P5**	**P6**	**P7**
G110		OA-4612	OA-4654	OA-3105		OA-4605	OA-3602
G113			OA-4204				
G114		OA-4602	OA-4501			OA-4203	OA-3104
G115	OA-4301	OA-3302	OA-4102	OA-3601	OA-4303	OA-3102	OA-3102
G117							
G118							
G122	OA-3900	OA-4655	OA-4302	OA-3102	OA-3610	OA-3200	OA-4301
G129	OA-4202						OA-3200
G130				OA-4604			
G133						OA-4910	

LIST OF REFERENCES

1. Nolen, Jefferey S. and Youngblood, Phillip D., "Naval Postgraduate School Scheduling Support System" Master's Thesis, Naval Posgraduate School, Monterey, California, March 1992.

2. Catalog Academic Year 1993, Naval Postgraduate School, 1993.

3. Anthony Brooke; David Kendrick; Alexander Meeraus, "GAMS: A User's Guide", Release 2.25, The Scientific Press, South San Francisco, CA, 1992.

4. Gerald G. Brown and Michael P. Olson "Elastic Modeling-with the X-System and GAMS", Naval Postgraduate School, v.93.01.25, 1993.

5. George L. Nemhauser and Laurence A. Wolsey "Integer and Combinational Optimization", John Wiley and Sons, New York, 1988.

6. Hammond, Tracy, Personal Communication, October 6 1993.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria VA 22304-6145	2
2.	Library, Code 052 Naval Postgraduate School Monterey CA 93943-5002	2
3.	Department Chairman, Code 55 Department of Operations Research Naval Postgraduate School Monterey, CA 93943-5002	1
4.	Prof. R. Kevin Wood, Code 55 Department of Operations Research Naval Postgraduate School Monterey, CA 93943-5002	4
5.	Prof. Richard Edwin Rosenthal, Code 55 Department of Operations Research Naval Postgraduate School Monterey, CA 93943-5002	1
6.	Library National Defence Management College Chung-Huo Taipei, Taiwan, R.O.C.	2
7.	Major Wu Hsi-Hsien National Defence Management College Chung-Huo Taipei, Taiwan, R.O.C.	2
8.	Informations Systems, Code 146A Naval Postgraduate School Monterey, California 93943-5100	1